Pascal-2™

Oregon
Software

Oregon
Software

The software described by this publication is subject to change without notice. Oregon Software assumes no responsibility for the use or reliability of any of its software that is modified without the prior written consent of Oregon Software.

Oregon Software holds right, title, and interest in the software described herein. The software, or any copies thereof, may not be made available to or distributed to any person or installation without the written approval of Oregon Software.

This publication, or parts of it, may be copied for use with the licensed software described herein, provided that all copies include this notice and all copyright notices.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in sub-division (b)(3)(ii) of the Rights in Technical Data and Computer Software clause at 52.227-7013 of the Federal Acquisitions Regulations (FARs).

Name of Contractor and Address:

Oregon Software, Inc.
6915 S.W. Macadam Avenue
Portland, Oregon 97219
Phone: 503-245-2202

© 1985, 1984, 1983 Oregon Software, Inc. All Rights Reserved. Printed in USA, January 1985.

Pascal-1, Pascal-2 and Oregon Software are trademarks of Oregon Software, Inc.

DEC, PDP, RSX, RSTS and RT-11 are trademarks of Digital Equipment Corporation.

TEX is a trademark of the American Mathematical Society.

TSX-Plus is a trade mark of S&H Computer Systems, Inc.

Updated January 1985

Pascal-2®

Version 2.1 for RT-11

User Manual

Second Edition
August 1983

Oregon
Software

UPDATE No. 2

Pascal–2 V2.1 User Manual

Second Edition

January 1985

Synopsis: This update package contains pages to be inserted in the *Pascal–2 User Manual* for RT. These new pages reflect the status of the software in Version 2.1D. This update notice should be kept in the front of your manual as a history of changes to the manual.

©1985 by Oregon Software, Inc.

INSTRUCTIONS

New and updated pages are indicated by the date at the bottom of the page. The nature of the changes, i.e., their substance, is indicated by the summary in the table below.

Remove the outdated pages of the user manual and insert the enclosed pages as listed below.

Remove	Insert	Purpose
Title page, ii	Title page, ii	Replaced copyright message.
iii to viii	iii to viii	Replaced the old table of contents with a new one.
xi	xi	Changed name and address in Preface.
1-1, 1-2	1-1, 1-2	Replaced reference to Jensen and Wirth's *Pascal User Manual and Report* with Doug Cooper's *Standard Pascal User Reference Manual.*
2-1, 2-2	2-1, 2-2	Added sentence explaining that your system manager may change the name of the `PASCAL` invocation command.
2-11 to 2-14.4	2-11 to 2-14.4	Replaced old "External Modules" with new "External Modules."
2-19, 2-20	2-19, 2-20	Corrected errors in the "Support Library" section.
2-29, 2-30	2-29, 2-30	Removed reference to `$$HEAP` in `NewOK` Listing.
2-43, 2-44	2-43, 2-44	Made minor editing changes.
2-49 to 2-56	2-49 to 2-56	Moved section "Calling FORTRAN Subroutines" to "External Modules."
3-9, 3-10	3-9, 3-10	Corrected lexical representation of *structured-constant.* Described prohibited use of *structured-constant* as case label.
3-17 to 3-20	3-17 to 3-20	Altered explanation of "Extended-Range Arithmetic." Corrected minor spelling and reference errors.
3-31, 3-32	3-31, 3-32	Corrected reference to **forward** as non-standard Pascal.
3-39, 3-40	3-39, 3-40	Corrected lexical representation of *structured-constant.*
4-1 to 4-30	4-1 to 4-36	Replaced old "Debugger Guide" with new "Debugger Guide."
5-17 to 5-20	5-17 to 5-20	Changed description of *max-len.* Changed some procedure and function names.
5-49, 5-50	5-49, 5-50	Corrected default value of L.
Index(all)	Index(all)	Added new entries.

Contents

Page

Preface . . . xi
Thanks to... . . . xi

Pascal–2 V2.1/RT-11 Introduction . . . xiii
About Version 2.1 . . . xiv
Pascal–2 Documentation Package . . . xv
Style Notes . . . xvi
Support Policy . . . xvii

Pascal–2 V2.1/RT-11 User Guide . . . 1-1
Introduction to the User Guide . . . 1-1
Getting Started . . . 1-1
Compiling the Program . . . 1-1
Checking For Errors . . . 1-2
Errors Detected at Run-Time . . . 1-3
Compilation Options . . . 1-4
The Program Listing . . . 1-4
The Formatter . . . 1-4
The Debugger . . . 1-6
Double Precision . . . 1-8
The Profiler . . . 1-8
Your Next Step . . . 1-9

Pascal–2 V2.1/RT-11 Programmer Reference . . . 2-1
Introduction . . . 2-1
Compiler Commands . . . 2-1
Compilation Switches . . . 2-2
Program Options . . . 2-2
Compiler Options . . . 2-2
Code Switches . . . 2-3
Checking Switches . . . 2-3
Processor Switches . . . 2-3
Embedded Switches . . . 2-4
Program Options . . . 2-5
Compiler Options . . . 2-5
Run-Time Checking Switches . . . 2-7
Compilation Examples . . . 2-7
Linking and Executing . . . 2-8
I/O Control Switches . . . 2-9
External Modules . . . 2-11
Calls to Pascal–2 Routines . . . 2-12
Example Using External Directive . . . 2-12
Calls to Non-Pascal–2 Routines . . . 2-14
Calling MACRO Subroutines . . . 2-14
Calling FORTRAN Subroutines . . . 2-14.2
External Module Libraries . . . 2-14.4
The Linker, Overlays, and the Librarian . . . 2-15
The Linker . . . 2-15
Virtual Jobs and the XM Monitor . . . 2-15
Overlays . . . 2-16
Low Memory Overlays . . . 2-16

    - Extended Memory Overlays 2-17
    - Linking a Program Whose Root Exceeds 16K 2-17
  - The Librarian 2-18
- Extended Precision 2-18
- Support Library 2-19
  - Initializing the Support Library 2-19
  - Support Library Data Definitions 2-19
  - Support Library's Use of Channels 2-21
- Run-Time Organization 2-22
  - Form of the Generated Code 2-22
  - Memory Organization 2-23
    - RT-11 System Area 2-23
    - Program Code 2-23
    - Global Variables 2-23
    - Constants 2-23
    - Tables 2-23
    - Run-Time Library 2-23
    - The Stack 2-24
    - The Heap 2-24
    - The Monitor and I/O Page 2-25
  - The Stack Frame 2-25
    - Function Return Value 2-26
    - Parameters 2-26
    - Return Link 2-26
    - Dynamic Link 2-26
    - Local Variables 2-26
    - Register Save Area 2-26
    - Temporary Storage 2-26
- Monitoring Memory Usage 2-27
  - The 'Space' Function 2-28
  - Function 'P$inew' and Procedure 'P$dispose' 2-28
  - Example: Function 'NewOK' 2-30
- Storage Allocation 2-33
- Run-Time Error Reporting 2-35
  - Examples 2-36
  - I/O Error Trapping 2-37
    - Procedure 'SayErr' 2-38.1
  - Customizing Error Reporting 2-39
- Error Termination Status 2-43
- Implementation Notes 2-44
  - Multiple Source Files 2-44
  - Local Files Closed on Procedure Exit 2-45
  - Specifying the Location of the Compiler's Work Files 2-46
  - Variable Initialization 2-46
  - Reading Command Lines 2-46
  - Foreground Operation 2-49
  - Lazy I/O 2-50
    - Incorporating Lazy I/O into V2.0 Programs 2-51
  - Terminal I/O 2-51
    - Single-Character or 'ODT' Mode 2-52
    - Function 'ReadSn' 2-53
  - Random Access to 'Text' Files 2-54
    - Procedure 'GetPos' 2-54
    - Procedure 'SetPos' 2-55

Unsigned Integer Conversion . . . 2-57
Compiler Optimizations . . . 2-59
Variable Assignments to Registers . . . 2-59
Assignment of Constants and Addresses to Registers . . . 2-59
Constant Folding . . . 2-59
Dead Code Elimination . . . 2-59
Boolean Expression Optimization . . . 2-59
Expression Targeting . . . 2-60
Common Subexpression Elimination . . . 2-60
Common Branch Tail Elimination . . . 2-60
Array Index Simplification . . . 2-60
Appendix A: Compilation Error Messages . . . 2-61
Appendix B: Run-Time Error Messages . . . 2-72
Appendix C: Compiler Errors . . . 2-76
Overflow Errors . . . 2-76
Consistency Checks . . . 2-76
Appendix D: Default File Extensions . . . 2-77
Appendix E: Entry Points in the Pascal Support Library . . . 2-78

Pascal–2 V2.1/RT-11 Language Specification . . . 3-1
Introduction to the Language Specification . . . 3-1
Changes in the Standard . . . 3-1
'For' Statement Control Variables . . . 3-1
File Declaration . . . 3-1
Parameter Compatibility . . . 3-1
Procedure and Function Parameters . . . 3-2
Conformant Array Parameters . . . 3-2
Literal Strings . . . 3-4
'Write,' 'Writeln' of 'Packed Array of Char' . . . 3-4
Identifiers . . . 3-5
Alternate Symbol Representations . . . 3-5
Implementation Definitions . . . 3-6
Standard Type 'Integer' . . . 3-6
Standard Type 'Real' . . . 3-6
Standard Type 'Char' . . . 3-6
Standard Type 'Text' . . . 3-6
'Set' Types . . . 3-6
I/O Definitions . . . 3-7
Syntax Extensions . . . 3-8
Identifiers . . . 3-8
Program Heading . . . 3-8
Declaration Order . . . 3-8
'%Include' Lexical Directive . . . 3-8
'%Page' Lexical Directive . . . 3-9
'External' and 'NonPascal' Directives . . . 3-9
Structured Constants . . . 3-9
Examples of Structured Constants . . . 3-10
Default Case Label ('Otherwise') . . . 3-13
I/O Support Extensions . . . 3-14
External File Access . . . 3-14
'Close' Procedure . . . 3-15
Random Access to Data Files ('Seek') . . . 3-15
String Input ('Read' and 'Readln') . . . 3-16
'Break' Procedure . . . 3-16
Octal Output . . . 3-16

    - Real Number Formatting 3-17
  - Low-Level Interface 3-18
    - Boolean Operators on Integer 3-18
    - Nondecimal Integer Constants 3-18
    - Extended-Range Arithmetic 3-18
    - "Origin" Declaration 3-20
    - 'Ref' Function 3-21
    - 'Size' and 'Bitsize' Functions 3-21
    - 'Loophole' Function 3-21
  - Non-Standard Language Elements 3-25
    - Program Parameters 3-25
    - Directives 3-25
    - 'Mod' of Negative Numbers 3-25
    - 'Eof' Not Accurate for Binary Files 3-25
    - Returning of Structured Types 3-25
  - Additional Predefined Functions and Procedures 3-26
    - Procedure 'Delete' 3-26
    - Procedure 'Rename' 3-26
    - Predefined Function 'Time' 3-27
    - Procedure 'TimeStamp' 3-28
  - Error Handling 3-30
    - Detected Errors 3-30
    - Undetected Errors 3-31
  - Appendix A: Predefined Identifiers 3-32
  - Appendix B: Reserved Words 3-32
  - Appendix C: Pascal–2 Syntax 3-33
    - Pascal–2 Syntax Diagrams 3-33
    - Extended Backus-Naur Form 3-38
    - Pascal–2 Lexical Description 3-39
    - Pascal–2 EBNF Syntax 3-40

- Pascal–2 V2.1/RT-11 Debugger Guide 4-1
  - Including the Pascal–2 Debugger in Your Program 4-2
      - Identifying Pascal Statements 4-2
  - Controlling the Debugger 4-4
      - Command Syntax 4-4
      - Exiting and Stopping the Debugger 4-5
      - Selective Debugging 4-5
  - Breakpoint Commands 4-6
    - B, B(): Control Breakpoints 4-6
    - K, K(): Killing of Breakpoints 4-6
    - V, V(): Data Breakpoints (Variables) 4-7
  - Execution Control Commands 4-8
    - G: Go 4-8
    - C, C(): Continue Execution 4-8
    - Examples of the B, K, D, G and C Commands 4-8
    - S, S(): Step to Next Statement 4-9
    - P, P(): Proceed to Next Statement 4-9
    - Examples of the S and P Commands 4-10
  - Tracking Commands 4-11
    - H, H(): History of Program Execution 4-11
    - T(): Execution Trace 4-11
    - Example of the T Command 4-11
  - Data Commands 4-13
    - W(): Write Variable Value 4-13

Variable Assignment . . . 4-14
Examples of the W Command and Variable Assignment . . . 4-15
Informational Commands . . . 4-17
D: Display Parameters . . . 4-17
L, L(): List Source Lines . . . 4-17
Utility Commands . . . 4-18
M(): Define Macro . . . 4-18
X(): Execute Macro . . . 4-18
Examples of the M and X Commands . . . 4-19
Execution Stack Commands . . . 4-20
H, H(): History of Program Execution . . . 4-20
N, N(): Names of Variables . . . 4-21
E(): Enter Stack-Frame Context . . . 4-21
Examples of the H, N, and E Commands . . . 4-22
Stepping Through a Debugger Session . . . 4-23
Debugging External Modules . . . 4-26
Differences in the Commands . . . 4-26
Example . . . 4-27
Overlays . . . 4-30
Appendix A: Debugger Command Summary . . . 4-31

The Pascal–2 Profiler . . . 4-32

Pascal–2 V2.1/RT-11 Utilities Guide . . . 5-1
Introduction to the Utilities Guide . . . 5-1

PASMAT: A Pascal–2 Formatter . . . 5-2
Overview of Capabilities . . . 5-2
Comments . . . 5-2
Statement Bunching . . . 5-3
Tables . . . 5-3
Using PASMAT . . . 5-3
Formatting Directives . . . 5-4
Limitations and Errors . . . 5-6
PASMAT Examples . . . 5-7

PB: A Pascal–2 Formatter . . . 5-9
Using PB . . . 5-9
Example . . . 5-10
Detailed Formatting Rules . . . 5-12

XREF: A Pascal–2 Cross-Reference Lister . . . 5-13
Using XREF . . . 5-13
Limitations . . . 5-13
Example of XREF Listing . . . 5-14

PROCREF: Pascal–2 Procedural Cross-Reference Lister . . . 5-15
Using PROCREF . . . 5-15
Limitations . . . 5-16
Example . . . 5-16

Dynamic String Package . . . 5-18
The Procedures and Functions . . . 5-19
Example . . . 5-20

MACRO-11 Procedures With Pascal–2 . . . 5-23
Design of MACRO-11 Procedures . . . 5-23
The PASMAC Macro Package . . . 5-24

- Using PASMAC . . . 5-25
- Procedure Definition Macros . . . 5-26
  - The 'Proc' Macro . . . 5-26
  - The 'Func' Macro . . . 5-27
  - The 'Param' Macro . . . 5-27
  - The 'Var' Macro . . . 5-28
  - The 'Save' Macro . . . 5-28
  - The 'Rsave' Macro . . . 5-29
  - The 'Begin' Macro . . . 5-29
  - The 'Endpr' Macro . . . 5-30
- Type Definitions . . . 5-31
- Example . . . 5-33
- Placing PASMAC into the System Macro Library . . . 5-36

PROSE: A Text Formatter . . . 5-38

- PROSE Basics . . . 5-39
- Structure of Directive Lines . . . 5-40
- Placement of Directives . . . 5-40
- Running the PROSE Program . . . 5-42
  - Header Files . . . 5-42
- Controlling Input to PROSE . . . 5-43
  - INPUT Directive . . . 5-43
  - OPTION Directive . . . 5-44
- Setting Up the Document's Format . . . 5-47
- Page Format . . . 5-47
  - FORM Directive . . . 5-47
  - Page Breaks . . . 5-48
  - Margins . . . 5-49
  - Paragraphs . . . 5-49
  - Comments . . . 5-51
- Changing Format Within the Text . . . 5-51
  - Breaking and Skipping Lines . . . 5-51
  - Keep Buffers . . . 5-52
  - Reset Directive . . . 5-53
- Creating an Index . . . 5-54
- Printing the Document . . . 5-55
  - Specifying Output Devices . . . 5-55
  - Printing Selected Pages and Sections . . . 5-56
- Appendix A: Examples of PROSE Directives in Text . . . 5-57
- Appendix B: Summary Directive Table . . . 5-63
- Appendix C: Historical Notes . . . 5-64

For More Information . . . Info-1

Index . . . Index-1

List of Figures

Figure **Page**

-- The Pascal–2 Software Development System xiii

2-1 Typical Memory Layout of a Pascal Program 2-24

2-2 Format of a Stack Frame 2-25

2-3 Tracking Memory Usage with the SPACE Function 2-29

Preface

This edition of the *Pascal–2 User Manual* for the RT-11 operating system corresponds to the latest software release and describes all of the features of Version 2.1 of the Pascal–2 software. This manual is printed looseleaf to allow us to send future errata as Update Packages with "change pages" to be inserted in the existing manuals. Update Packages are sent with new releases when changes made to the software require new or different explanations in the manual. New releases of software, including a set of release notes, are sent to licensed users of the Pascal–2 compiler approximately every six months.

We are grateful to the many readers whose thoughtful and perceptive suggestions have helped us improve our manuals. Please fill out and return the evaluation report provided at the end of this manual, or address your comments directly to:

David Spencer, Manager
Technical Publications
Oregon Software
6915 SW Macadam Ave.
Portland, Oregon 97219

We appreciate hearing from you.

Thanks to...

Oregon Software got its start at OMSI—the Oregon Museum of Science and Industry. OMSI is a private educational organization chartered "to enhance the general public understanding of science and technology, with a strong commitment to education," particularly of youth. It was in the Research Laboratory at OMSI that we began writing software. Seven of us came from OMSI to found Oregon Software in September, 1977. Because of the close association, the name "OMSI" stayed with us for a while, and we continue to support OMSI and its educational programs.

As part of our own research, we've been using and distributing TEX, a text-processing system developed by Don Knuth at Stanford University. This publication is typeset in the Computer Modern Roman family of type faces with the TEX system. Draft versions were produced at Oregon Software on an Imprint-10 laser printer driven by TEX-in-Pascal and our VAX-11/780.

Special thanks to Barbara Beeton and Monty Nichols for their encouragement and assistance with TEX and to David Kellerman and Barry Smith for implementing TEX at Oregon Software.